



BAB II

TINJAUAN PUSTAKA

2.1. Teori Umum

2.1.1. Pengertian Komputer

Asropudin (2013:19) mengemukakan, “Komputer adalah alat bantu pemrosesan data secara elektronik dan cara pemrosesan datanya berdasarkan urutan instruksi atau program yang tersimpan dalam memori masing-masing komputer”.

Hartono (2013:27) mengemukakan, “Komputer adalah sebuah mesin yang dapat dikendalikan melalui perintah (*programmable machine*) yang dirancang untuk secara otomatis melakukan serangkaian urutan perhitungan (*arithmetic*) atau proses-proses yang diurutkan secara logis”.

Wahyudi (2012:3) menyatakan, “Komputer adalah Peralatan (*device*) yang menerima data (*input*) dan menyimpan (*storage*) kemudian di proses (*process*) untuk menghasilkan data dalam bentuk lain (*output*).”

Jadi, komputer adalah alat pemrosesan data elektronik yang membantu melakukan perintah-perintah perhitungan atau proses-proses tertentu yang telah diurutkan atau diprogramkan.

2.1.2. Pengertian Perangkat Lunak

Sukanto dan Shalahuddin (2013:2), “Perangkat Lunak (*Software*) adalah program komputer yang terasosiasi dengan dokumen perangkat lunak seperti dokumentasi kebutuhan, model desain dan cara penggunaan (*user manual*).”

Sutanta (2005:20), “*Software* merupakan serangkaian intruksi dengan aturan tertentu yang mengatur operasi perangkat keras.”

Jadi, perangkat lunak atau *software* merupakan program komputer terasosiasi dengan dokumen perangkat lunak yang memiliki serangkaian aturan tertentu dalam pengoperasiannya.



2.1.3. Pengertian Program

Sutarman (2012:3), “Program adalah barisan perintah atau instruksi yang disusun sehingga dapat dipahami oleh komputer dan kemudian dijalankan sebagai barisan perhitungan numerik, dimana barisan perintah tersebut berhingga, berakhir, dan menghasilkan *output*.”

Sutanta (2005:21), “Program merupakan bahasa yang mudah dipahami oleh orang awam.”

Jadi, program merupakan kumpulan instruksi yang dapat dimengerti komputer dan orang awam yang menghasilkan *output*.

2.1.4. Pengertian Data

Asropudin (2013:22), “Data adalah kumpulan dari angka-angka maupun karakter-karakter yang tidak memiliki arti”.

Sutabri (2012:3), “Data adalah hal, peristiwa, atau kenyataan lain ataupun yang mengandung sesuatu pengetahuan untuk dijadikan dasar guna penyusunan keterangan, pembuatan kesimpulan, atau penetapan keputusan.”

Wahyudi (2012:3), “Data adalah suatu file ataupun field yang berupa karakter atau tulisan dan gambar.”

Jadi, data adalah kumpulan dari angka maupun karakter yang berasal dari kenyataan yang belum berarti dan harus diolah terlebih dahulu untuk menjadi suatu informasi yang lebih berarti bagi penggunanya.

2.1.5. Pengertian Pengolahan Data

Ladjmudin (2013:9) menjelaskan bahwa “Pengolahan Data adalah masa atau waktu yang digunakan untuk mendeskripsikan perubahan bentuk data menjadi informasi yang memiliki kegunaan.”

Sutarman (2012:4), “Pengolahan data adalah proses perhitungan atau transformasi data *input* menjadi informasi yang mudah dimengerti ataupun sesuai dengan yang diinginkan.”

Kristanto (2008:8) menjelaskan, “Pengolahan data adalah waktu yang digunakan untuk menggambarkan perubahan bentuk data menjadi informasi yang memiliki kegunaan.”



Jadi, pengolahan data merupakan proses pengolahan data yang diinputkan untuk diubah menjadi informasi yang berguna.

2.1.6. Pengertian Sistem

Sutabri (2012:10), “Sistem merupakan kumpulan atau himpunan dari unsur, komponen, atau variabel yang terorganisir, saling berinteraksi, saling tergantung satu sama lain, dan terpadu.”

Kristanto (2008:1), “Sistem adalah jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau menyelesaikan suatu sasaran tertentu.”

Jadi, sistem merupakan sekumpulan unsur atau elemen yang saling berkaitan dan saling mempengaruhi dalam melakukan kegiatan bersama untuk mencapai suatu tujuan.

2.1.7. Karakteristik Sistem

Sutabri (2012:20-21) menjelaskan bahwa sistem memiliki karakteristik atau sifat-sifat tertentu yang mencirikan bahwa hal tersebut bisa dikatakan sebagai suatu sistem. Adapun karakteristik yang dimaksud adalah sebagai berikut :

1. **Komponen Sistem (*Components*)**

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang bekerja sama membentuk suatu kesatuan.

2. **Batasan Sistem (*Boundary*)**

Ruang lingkup sistem merupakan daerah yang membatasi antara sistem dengan sistem yang lainnya atau sistem dengan lingkungan luarnya. Batasan sistem ini memungkinkan suatu sistem dipandang sebagai satu kesatuan yang tidak dapat dipisahkan.

3. **Lingkungan Luar Sistem (*Environment*)**

Bentuk apapun yang ada di luar ruang lingkup atau batasan sistem yang mempengaruhi operasi sistem tersebut disebut dengan lingkungan luar sistem.

4. **Penghubung Sistem (*Interface*)**

Media yang menghubungkan sistem dengan subsistem yang lain disebut dengan penghubung sistem atau *interface*.

5. **Masukan Sistem (*Input*)**

Energi yang dimasukkan ke dalam sistem disebut masukan sistem, yang dapat berupa pemeliharaan (*maintanace input*) dan sinyal (*signal input*).



6. Keluaran Sistem (*Output*)

Hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna. Keluaran ini merupakan masukan bagi sub sistem yang lain.

7. Pengolahan Sistem (*Process*)

Suatu sistem dapat mempunyai suatu proses yang akan mengubah masukan menjadi keluaran.

8. Sasaran Sistem (*Objective*)

Suatu sistem memiliki sasaran dan tujuan yang pasti dan bersifat deterministik. Kalau suatu sistem tidak memiliki sasaran, maka operasi sistem tidak ada gunanya. Suatu sistem dikatakan berhasil bila mengenai sasaran atau tujuan yang telah direncanakan.

2.1.8. Metode Pengembangan Sistem

Sukanto dan Shalahudin (2013:28) menjelaskan tentang metode pengembangan sistem yaitu *waterfall*. Metode air terjun (*waterfall*) sering juga disebut model sekuensial linier (*sequential linear*) atau alur hidup klasik (*classic life cycle*). Model air terjun menyediakan pendekatan alur hidup terurut mulai dari analisis, desain, pengodean, pengujian, dan pemeliharaan.

a. Analisis

Tahap analisis dilakukan secara intensif untuk menspesifikasikan kebutuhan sistem agar dapat dipahami sistem seperti apa yang dibutuhkan oleh *user*.

b. Desain

Tahap desain adalah proses multi langkah yang fokus pada desain pembuatan program sistem termasuk struktur data, arsitektur sistem, representasi antarmuka, dan prosedur pengkodean. Tahap ini mentranslasi kebutuhan sistem dari tahap analisis kebutuhan ke representasi desain agar dapat diimplementasikan menjadi program pada tahap selanjutnya.

c. Pengkodean

Pada tahap pengkodean, desain harus ditranslasikan ke dalam program sistem. Hasil dari tahap ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahap desain.

d. Pengujian

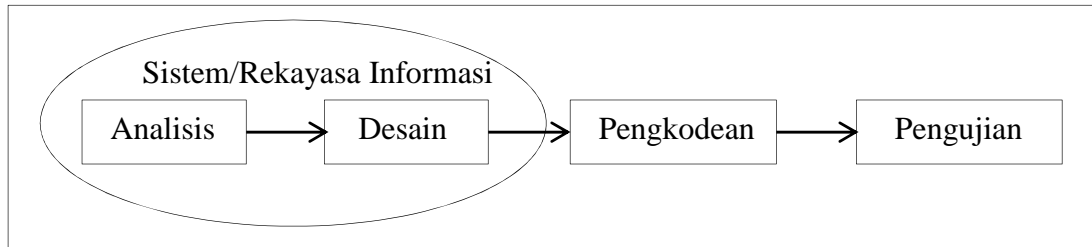
Tahap pengujian fokus pada sistem dari segi logika dan fungsional dan memastikan bahwa semua bagian sudah diuji. Hal ini dilakukan untuk meminimalisir kesalahan (*error*) dan memastikan keluaran yang dihasilkan sesuai dengan yang diinginkan.

e. Pemeliharaan

Tidak menutup kemungkinan sebuah sistem mengalami perubahan ketika sudah dikirimkan ke *user*. Perubahan bisa terjadi karena adanya kesalahan yang muncul dan tidak terdeteksi saat pengujian atau sistem harus beradaptasi dengan lingkungan baru. Tahap pemeliharaan dapat mengulangi proses pengembangan mulai dari



analisis spesifikasi untuk perubahan sistem yang sudah ada, tapi tidak untuk sistem baru.



(Sumber : Sukamto dan Shalahudin, 2013:28)

Gambar 2.1. Ilustrasi Model *Waterfall*

2.2. Teori Khusus

2.2.1. Pengertian *DFD* (Data Flow Diagram)

Sukamto dan Shalahuddin, (2013:70), “*Data Flow Diagram (DFD)* atau dalam bahasa indonesia menjadi Diagram Alir Data (DAD) adalah representasi grafik yang menggambarkan aliran informasi dan transformasi informasi yang diaplikasikan sebagai data yang mengalir dari masukan (*input*) dan keluaran (*output*).”

Ladjmudin (2013:64) mendefinisikan bahwa, “*DFD* adalah model dari sistem untuk menggambarkan pembagian sistem ke modul yang lebih kecil.”



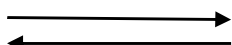
Kristanto (2008:61) menjelaskan “*Data Flow Diagram* merupakan suatu model logika data atau proses yang dibuat untuk menggambarkan darimana asal data dan kemana tujuan data yang keluar dari sistem, dimana data disimpan, proses apa yang menghasilkan data tersebut dan interaksi antara data yang tersimpan dan proses yang dikenakan pada data tersebut.”

Sukamto dan Shalahuddin, (2013:71), menjelaskan notasi pada *DFD* adalah sebagai berikut :

Tabel 2.1. Simbol-simbol *Data Flow Diagram*

No	Simbol	Keterangan
1.		Proses atau fungsi atau prosedur; pada pemodelan perangkat lunak yang akan diimplementasikan dengan pemrograman terstruktur, maka pemodelan notasi inilah yang harusnya menjadi fungsi atau prosedur di dalam kode program

Lanjutan Tabel 2.1. Simbol-simbol dalam *Data Flow Diagram*

No	Simbol	Keterangan
2.		<i>File</i> atau basis data atau penyimpanan (<i>storage</i>); pada pemodelan perangkat lunak yang akan diimplementasikan dengan pemrograman terstruktur, maka pemodelan notasi inilah yang harusnya dibuat menjadi tabel-tabel basis data yang dibutuhkan, tabel-tabel ini juga harus sesuai dengan perancangan tabel-tabel pada basis data (<i>Entity Relationship Diagram</i> (ERD), <i>Conceptual Data Model</i> (CDM), <i>Physical Data Model</i> (PDM))
3.		Entitas luar (<i>external entity</i>) atau masukan (<i>input</i>) atau keluaran (<i>output</i>) atau orang yang memakai/berinteraksi dengan perangkat lunak yang dimodelkan atau sistem lain yang terkait dengan aliran data dari sistem yang dimodelkan
4.		Aliran data; merupakan data yang dikirim antar proses, dari penyimpanan ke proses, atau dari proses ke masukan (<i>input</i>) atau keluaran (<i>output</i>)

(Sumber : Sukamto dan Shalahuddin, 2013:71)

Menurut Sukamto dan Shalahuddin (2013:72), berikut ini adalah tahapan-tahapan perancangan dengan menggunakan *Data Flow Diagram* :

1. Membuat *DFD* Level 0 atau sering disebut juga *Context Diagram*
DFD Level 0 menggambarkan sistem yang akan dibuat sebagai suatu entitas tunggal yang berinteraksi dengan orang maupun sistem lain. *DFD* Level 0 digunakan untuk menggambarkan interaksi antara sistem yang akan dikembangkan dengan entitas luar.
2. Membuat *DFD* Level 1
DFD Level 1 digunakan untuk menggambarkan modul-modul yang ada dalam sistem yang akan dikembangkan. *DFD* Level 1 merupakan hasil *breakdown DFD* Level 0 yang sebelumnya sudah dibuat.



3. Membuat *DFD* Level 2

Modul-modul pada *DFD* Level 1 dapat di-*breakdown* menjadi *DFD* Level 2. Modul mana saja yang harus di-*breakdown* lebih detail tergantung pada kedetailan modul tersebut. Apabila modul tersebut sudah cukup detail dan rinci maka modul tersebut sudah tidak perlu di-*breakdown* lagi. Untuk sebuah sistem, jumlah *DFD* Level 2 sama dengan jumlah modul pada *DFD* Level 1 yang di-*breakdown*.

4. Membuat *DFD* Level 3 dan seterusnya



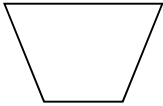

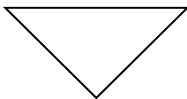
DFD Level 3,4,5, dan seterusnya merupakan *breakdown* dari modul pada *DFD* Level di-atasnya. *Breakdown* pada level 3,4,5, dan seterusnya aturannya sama persis dengan *DFD* Level 1 atau 2.

2.2.2. Pengertian Blockchart

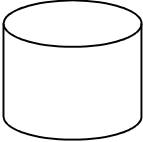

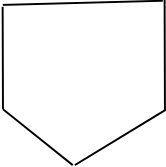
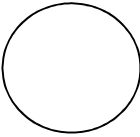

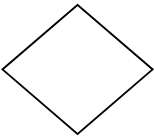

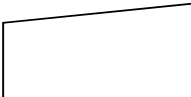
Kristanto (2008:75) menjelaskan, “*Blockchart* berfungsi untuk memodelkan masukan, keluaran, proses maupun transaksi dengan menggunakan simbol-simbol tertentu. Pembuatan *Blockchart* harus memudahkan bagi pemakai dalam memahami alur dari sistem atau transaksi.”

Kristanto (2008:75) menjelaskan, “Simbol-simbol yang sering digunakan dalam *blockchart* dapat dilihat pada tabel berikut ini:

Tabel 2.2. Simbol-simbol dalam *Block Chart*

No	Simbol	Keterangan
1.		Menandakan dokumen, bisa dalam bentuk surat, formulir, buku/bendel/berkas atau cetakan.
2.		Multi dokumen
3.		Proses Manual
4.		Proses dilakukan oleh komputer.
5.		Menandakan dokumen yang diarsipkan (arsip manual)

**Lanjutan Tabel 2.2.** Simbol-simbol dalam *Block Chart*

No	Simbol	Keterangan
6.		Data penyimpanan (<i>Storage</i>)
7.		Proses apa saja yang tidak <i>terDefinisi</i> termasuk aktivitas fisik.
8.		Terminasi yang mewakili simbol tertentu untuk digunakan pada aliran lain pada halaman yang lain.
9.		Terminal yang mewakili simbol tertentu untuk digunakan pada aliran lain pada halaman yang sama.
10.		Terminal yang menandakan awal dan akhir dari suatu aliran.
11.		Pengambilan keputusan (<i>Decision</i>).
12.		Layar peraga (<i>monitor</i>).
13.		Pemasukkan data secara manual.

(Sumber : Kristanto, 2008 : 75)

2.2.3. Pengertian ERD (*Entity Relational Diagram*)

Sukamto dan Shalahuddin, (2013:289), “*Entity Relationship Diagram* (ERD) adalah pemodelan awal basis data yang dikembangkan berdasarkan teori



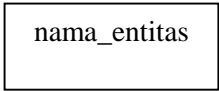
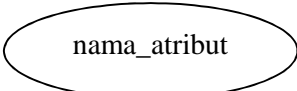
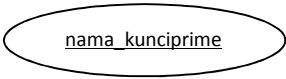

himpunan dalam bidang matematika untuk permodelan basis data *relational*.”

Ladjmudin (2013:142) mendefinisikan bahwa, “*ERD* adalah suatu model jaringan yang menggunakan susunan data yang disimpan dalam sistem secara abstrak.”


Al Fatta (2007:121) menjelaskan, “*ERD (Entity Relationship Diagram)* adalah gambar atau diagram yang menunjukkan informasi dibuat, disimpan, dan digunakan dalam sistem bisnis.”

Sukamto dan Shalahuddin, (2013:50), menjelaskan simbol-simbol yang digunakan dalam *ERD*, yaitu:

Tabel 2.3. Simbol-simbol *Entity Relationship Diagram (ERD)*

No	Simbol	Keterangan
1.	Entitas/ <i>entity</i> 	Entitas merupakan data inti yang akan disimpan; bakal tabel pada basis data; benda yang memiliki data dan harus disimpan datanya adar dapat diakses oleh aplikasi komputer; penamaan entitas biasanya lebih ke kata benda dan belum merupakan nama tabel.
2.	Atribut 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas.
3.	Atribut kunci primer 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas dan digunakan sebagai kunci akses <i>record</i> yang diinginkan; biasanya berupa id; kunci primer dapat lebih dari satu kolom, asalkan kombinasi dari beberapa kolom tersebut dapat bersifat unik (berbeda tanpa ada yang sama)
4.	Relasi 	Relasi yang menghubungkan antar entitas; biasanya diawali dengan kata kerja

**Lanjutan Tabel 2.3.** Simbol-simbol *Entity Relationship Diagram (ERD)*

No	Simbol	Keterangan
5.	Asosiasi / <i>association</i> 	Penghubung antara relasi dan entitas di mana di kedua ujungnya memiliki <i>multiplicity</i> kemungkinan jumlah pemakaian. Kemungkinan jumlah maksimum keterhubungan antara entitas satu dengan entitas yang lain disebut dengan kardinalitas. Misalkan ada kardinalitas 1 ke N atau sering disebut dengan <i>one to many</i> menghubungkan entitas A dan entitas B

(Sumber : Sukanto dan Shalahuddin, 2013:50)

2.2.4. Pengertian *Flowchart*


Ladjmudin (2013:263) mendefinisikan bahwa, “*Flowchart* adalah bagan-bagan yang mempunyai arus yang menggambarkan langkah-langkah penyelesaian suatu masalah.”

Ewolf Community (2012:16) mengemukakan, “*Flowchart* adalah simbol-simbol pekerjaan yang menunjukkan bagan aliran proses yang saling terhubung. Jadi, setiap simbol *flowchart* melambangkan pekerjaan dan instruksinya”.


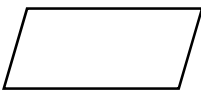
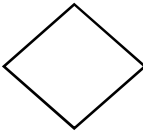
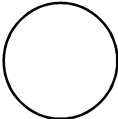
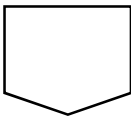
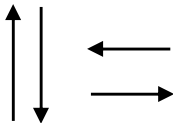

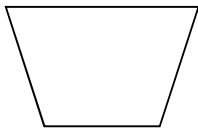


Dari kedua pengertian di atas, dapat disimpulkan bahwa *flowchart* merupakan sebuah diagram dengan simbol-simbol grafis yang menyatakan aliran atau proses yang menampilkan langkah-langkah yang disimbolkan dalam bentuk kotak, beserta urutannya.

Ewolf Community (2012:16) juga menjelaskan tentang simbol-simbol yang sering digunakan dalam *flowchart* dapat dilihat pada tabel berikut ini:


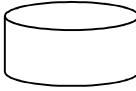
Tabel 2.4. Simbol-simbol dalam *Flow Chart*

No.	Simbol	Arti
1.		Simbol <i>Start</i> atau <i>End</i> yang mendefinisikan awal atau akhir dari sebuah <i>flowchart</i>

**Lanjutan Tabel 2.4.** Simbol-simbol dalam *Flow Chart*

No	Simbol	Arti
2.		Simbol pemrosesan yang terjadi pada sebuah alur kerja
3.		Simbol <i>Input/Output</i> yang mendefinisikan masukan dan keluaran proses
4.		Simbol untuk memutuskan proses lanjutan dari kondisi tertentu.
5.		Simbol konektor untuk menyambung proses pada lembar kerja yang sama
6.		Simbol konektor untuk menyambung proses pada lembar kerja yang berbeda
7.		Simbol untuk menghubungkan antar proses atau antar symbol
8.		Simbol yang menyatakan piranti keluaran, seperti layar monitor, <i>printer</i> , dll
9.		Simbol yang mendefinisikan proses yang dilakukan secara manual
10.		Simbol masukan atau keluaran dari atau ke sebuah dokumen
11.		Simbol yang menyatakan bagian dari program (sub program)

**Lanjutan Tabel 2.4.** Simbol-simbol dalam *Flow Chart*

No	Simbol	Arti
12.		Simbol masukan atau keluaran dari atau ke sebuah pita <i>magnetic</i>
13.		Simbol <i>database</i> atau basis data

(Sumber : Ewolf community, 2012:16)

2.2.5. Pengertian Kamus Data

Sukamto dan Shalahuddin, (2013:73), “Kamus data adalah kumpulan daftar elemen data yang mengalir pada sistem perangkat lunak sehingga masukan (*input*) dan keluaran (*output*) dapat dipahami secara umum (memiliki standar cara penulisan).”

Sukamto dan Shalahuddin, (2013:73), menjelaskan simbol-simbol yang di gunakan dalam kamus data, yaitu :

Tabel 2.5. Simbol-simbol dalam Kamus Data

No	Simbol	Arti
1	=	disusun atau terdiri atas
2	+	Dan
3	[]	baik ...atau...
4	{ } ⁿ	n kali diulang/ bernilai banyak
5	()	data operasional
6	*...*	batas komentar

(Sumber : Sukamto dan Shalahuddin, 2013:73)

2.2.6. Pengertian Daftar Kejadian (*Event List*)

Kristanto (2008:70) menjelaskan bahwa daftar kejadian (*event list*) digambarkan dalam bentuk kalimat sederhana dan berfungsi untuk memodelkan kejadian yang terjadi dalam lingkungan sehari-hari dan membutuhkan tanggapan atau respon dari sistem. Adapun cara mendeskripsikan daftar kejadian adalah sebagai berikut:

1. Pelaku adalah *entity* luar, jadi bukan sistem.



2. Menguji setiap *entity* luar dan mencoba mengevaluasi setiap *entity* luar yang terjadi pada sistem.
3. Hati-hati dengan kejadian yang spesifik, yang tak sengaja menyatu dalam paket yang sama.
4. Harus diingat bahwa kejadian yang dimodelkan bukan hanya interaksi normal antara sistem dengan *entity* luar, karena itu harus dievaluasi kebutuhan sistem untuk menanggapi kejadian yang gagal.
5. Setiap aliran keluaran sebaiknya merupakan respon dari kejadian.
6. Setiap kejadian yang tidak berorientasi pada waktu dalam daftar kejadian sebaiknya mempunyai masukan sehingga sistem dapat mendeteksi kejadian yang berlangsung.
7. Setiap kejadian sebaiknya menghasilkan keluaran langsung sebagai respon atau disimpan dalam berkas untuk bahan masukan.

2.3. Pengertian-Pengertian Judul

2.3.1. Pengertian Aplikasi

Asropuddin (2013:6), “Aplikasi adalah *software* yang dibuat oleh suatu perusahaan komputer untuk mengerjakan tugas-tugas tertentu, misalnya *Ms-Word*, *Ms-Excel*.”

Sutabri (2012:147), “Aplikasi adalah alat terapan yang difungsikan secara khusus dan terpadu sesuai kemampuan yang dimilikinya.”

Jadi, dapat disimpulkan bahwa aplikasi adalah alat yang dibuat untuk mengerjakan tugas-tugas tertentu yang difungsikan secara terpadu.

2.3.2. Pengertian Pencatatan

Mursyidi (2010:18), “Pencatatan atau *Recording* merupakan hal yang harus ditunjang oleh dokumen sumbernya (nota, faktur, bukti memorial, dan lain-lain).”

(Sumber: www.bambanghariyanto.com/2012/06/proses-akuntansi-menurut-para-ahli.html?m=1)

2.3.3. Pengertian Data

Sutabri (2012:3), “Data adalah hal, peristiwa, atau kenyataan lain ataupun yang mengandung sesuatu pengetahuan untuk dijadikan dasar guna penyusunan keterangan, pembuatan kesimpulan, atau penetapan keputusan.”



2.3.4. Pengertian Pembelian

Bodnar dan Hopwood (2001:323), “Pembelian merupakan proses bisnis dalam memilih sumber daya, pemesanan dan perolehan barang atau jasa.”

(Sumber: blogdee.blogspot.com/2011/10/pembelian.html?m=1)

2.3.5. Pengertian Tiket Pesawat

MENHUB (2011:3), “Tiket Pesawat adalah dokumen berbentuk cetak, melalui proses elektronik atau bentuk lainnya, yang merupakan salah satu bukti adanya perjanjian angkutan udara antara penumpang dan pengangkut, dan hak penumpang untuk menggunakan pesawat udara atau diangkut dengan pesawat udara.”

(Sumber: hubud.dephub.go.id)

2.3.6. Pengertian Kargo

IATA (2005), “Kargo adalah semua barang yang akan diangkut melalui darat, laut, maupun udara dengan menggunakan Surat Muatan Udara (SMU) tetapi tidak termasuk pos atau barang lain yang dimuat dalam perjanjian konvensi pos internasional dan bagasi yang disertai tiket penumpang.”

(Sumber: niconadirsyah.blospot.com/2011_11_15_archive.html?m=1)

2.3.7. Pengertian Travel Grand Puri Tiara Palembang

Travel Grand Puri Tiara Palembang merupakan sarana layanan umum yang bergerak di bidang jasa travel tiket pesawat dan pengiriman kargo yang berlokasi di Jl. Kebun Bunga A8, Komplek Villa Angkasa Permai Km.9 Palembang. Travel Grand Puri Tiara telah bekerja sama dengan berbagai *airlines* atau maskapai penerbangan, seperti Garuda Indonesia, Sriwijaya Air, Lion Air, Citylink dan Batik Air. Tiket pesawat pada Travel Grand Puri Tiara meliputi tiket pesawat *regular* dan tiket pesawat *tour* untuk berbagai tujuan baik dalam negeri maupun luar negeri. Sedangkan untuk pengiriman kargo pada Travel Grand Puri Tiara dapat berupa dokumen, obat-obatan, hingga sepeda motor yang merupakan perpanjangan tangan dari Mega Kargo Jakarta dan Queen Express Jakarta untuk daerah pengiriman Sumatera Selatan. Travel ini menjalankan kegiatan bisnisnya



berdasarkan prinsip-prinsip tata kelola yang baik sehingga dapat berdaya saing yang tinggi dalam era globalisasi.

2.3.8. Pengertian Aplikasi Pencatatan Data Pembelian Tiket Pesawat Dan Kargo Pada Travel Grand Puri Tiara Palembang

Aplikasi Pencatatan Data Pembelian Tiket Pesawat Dan Kargo Pada Travel Grand Puri Tiara Palembang merupakan alat yang dibuat untuk membantu karyawan maupun karyawan dalam melakukan proses pencatatan data pembelian tiket pesawat dan pengiriman kargo pada Travel Grand Puri Tiara Palembang agar lebih efektif dan efisien serta agar memudahkan dalam pembuatan laporan bulanan.

2.4. Teori Program

2.4.1. Basis Data (*Database*)

Priyadi (2014:2) menyatakan, “Secara umum, pengertian basis data adalah sekumpulan fakta berupa representasi tabel yang saling berhubungan dan disimpan dalam media penyimpanan secara digital.”

Kristanto (2008:79), “Basis data adalah kumpulan data yang dapat digambarkan sebagai aktifitas dari satu atau lebih organisasi yang berelasi.”

Priyadi (2014:3) juga menjelaskan beberapa elemen basis data, diantaranya adalah sebagai berikut:

1. *Tabel*
Pada suatu basis data, tabel direpresentasikan menjadi suatu bentuk segiempat berupa matriks, yang terdiri dari kolom dan baris.
2. *Field*
Pada tabel basis data, kolom merupakan representasi untuk sebuah nama *field* yang pembacaan datanya dilakukan secara vertikal.
3. *Record*
Pada tabel basis data, baris merupakan suatu representasi untuk sebuah *record* yang pembacaan datanya dilakukan secara horizontal. Satu baris pada sebuah tabel merupakan data yang dimiliki oleh satu *record*. Nilai-nilai yang dimiliki oleh sebuah *record* merupakan gabungan dari semua *field* yang terdapat dalam tabel tersebut.



4. Kardinalitas

Kardinalitas merupakan batasan dari banyaknya hubungan, yang dapat dilakukan oleh himpunan entitas dalam melakukan relasi dengan himpunan entitas lainnya. Variasi kemungkinan untuk melakukan relasi yang dimiliki oleh kardinalitas terdiri dari empat macam, yaitu:

- a. Satu ke satu \longrightarrow (1:1).
- b. Satu ke banyak \longrightarrow (1:N).
- c. Banyak ke satu \longrightarrow (N:1).
- d. Banyak ke banyak \longrightarrow (N:N).

2.4.2. Pengertian *HTML*

Winarno,dkk (2014:1), “*Hypertext Markup Language (HTML)* adalah sebuah bahasa untuk menampilkan konten di Web.”

Saputra (2013:1), “HTML merupakan bahasa pemrograman yang digunakan untuk membangun kerangka atau pondasi *web*.”

Anhar (2010:40), “HTML adalah sekumpulan simbol-simbol atau *tag-tag* yang dituliskan dalam sebuah *file* untuk menampilkan halaman pada *web browser*.”

2.4.3. Pengertian *MySQL*

Saputra (2013:14) menyatakan, “*MySQL* merupakan *database storage engine* yang paling banyak digunakan oleh *web developer* karena sifatnya *free*, alias gratis.”

Anhar (2010:21), “*MySQL* adalah sebuah perangkat lunak sistem manajemen basis data *SQL* atau *DBMS* dan merupakan *multithread*, *multi-user* yang bersifat gratis dibawah lisensi *GNU General Public Licence (GPL)*.”

2.4.4. Sekilas Tentang *PHP*

2.4.4.1. Pengertian *PHP (Hypertext Preprocessor)*

Sidik (2012:4), “*PHP* merupakan secara umum dikenal sebagai bahasa pemrograman *script-script* yang membuat dokumen *HTML* secara *on the fly* yang dieksekusi di server web, dokumen *HTML* yang dihasilkan dari suatu aplikasi



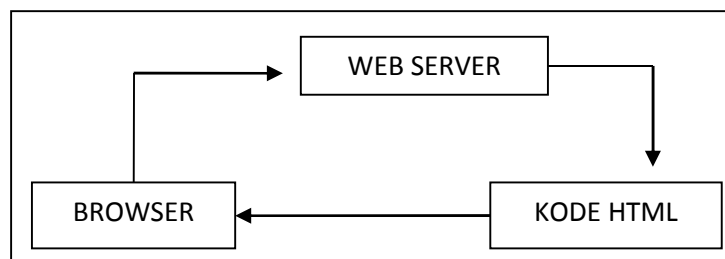
bukan dokumen *HTML* yang dibuat dengan menggunakan editor teks atau editor *HTML*.”

Winarno,dkk (2014:49), “*PHP* atau *PHP Hypertext Preprocessor* adalah sebuah bahasa pemrograman web berbasis *server* (*server-side*) yang mampu memarsing kode *PHP* dari kode web dengan ekstensi *.php*, sehingga menghasilkan tampilan *website* yang dinamis di sisi *client* (*browser*).”

2.4.4.2. Konsep Kerja *PHP*

Saputra (2013:4) menjelaskan tentang konsep kerja *PHP* adalah sebagai berikut:

- Server membaca permintaan dari *client/browser*.
- Kemudian dilanjutkan untuk mencari halaman/*page* pada *server*.
- Server* melakukan instruksi yang diberikan oleh *PHP* untuk melakukan modifikasi pada halaman/*page*.
- Selanjutnya hasil modifikasi tersebut akan dikembangkan kepada *client/browser*.



(Sumber : Saputra, 2013:5)

Gambar 2.2. Cara Kerja *PHP*

2.4.4.3. Skrip *PHP*

Sidik (2012:9) mengemukakan bahwa, “*PHP* banyak digunakan karena kemudahannya. Pengembangan aplikasi web dapat dengan mudah untuk menyisipkan script atau menghasilkan dokumen *HTML* dengan menggunakan *HTML*”.

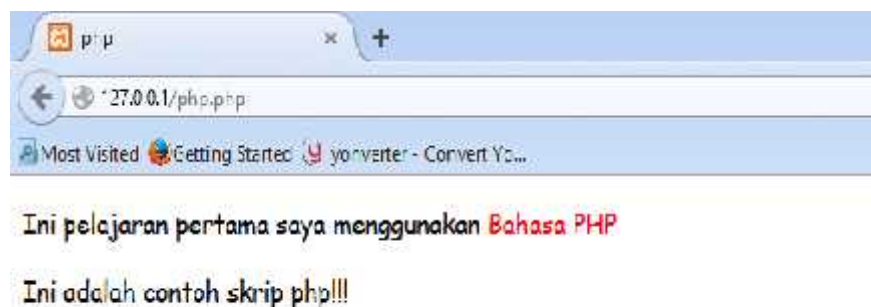
Berikut adalah contoh yang umum digunakan untuk menjelaskan tentang *PHP* sebagai *script* yang disisipkan (*embedded script*) dalam dokumen *HTML* :



```
<!doctype html>
<html lang="en">
  <head>
    <title>php</title>
  </head>
  <body>
    <p>Ini pelajaran pertama saya
    <?php
      //kode yang digunakan untuk mencetak/ menampilkan data
      echo"menggunakan <font color='red'>Bahasa PHP</font>";

    ?>
  </p>
  <p>
    Ini adalah contoh skrip php!!!
  </p>
</body>
</html>
```

Bila dijalankan melalui *browser*, kode tersebut membentuk tampilan seperti berikut:



Gambar 2.3. Tampilan *PHP* Sederhana Pada Browser

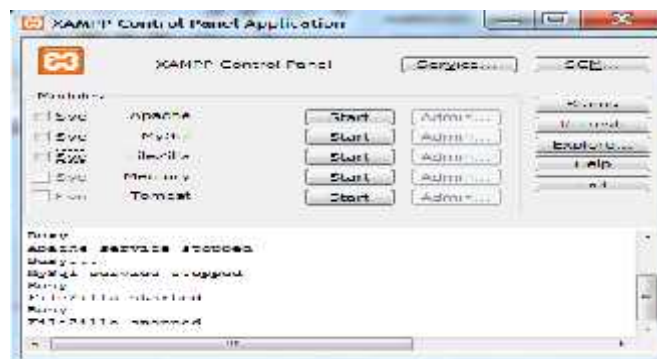
2.4.5. Sekilas tentang *XAMPP*

Sidik (2012:72), “*XAMPP* (X/Window/Linux) Apache MySQL PHP dan Perl) merupakan paket *server web* PHP dan database MySQL yang paling populer di kalangan pengembangan *web* dengan menggunakan PHP dan MySQL sebagai *databasenya*. Paket *XAMPP*, sesuai dengan kepanjangannya, X yang berarti *Windows* atau *Linux*, pengguna bisa memilih paket yang diinginkan untuk *Windows* atau *Linux*. *XAMPP* dapat diperoleh dari <http://xampp.org> atau <http://apachefriends.org>.



Menurut Sidik (2012:73), langkah-langkah pemasangan XAMPP dari file zip yang telah di download adalah sebagai berikut :

1. Unzip file XAMPP yang telah didownload, ke dalam file directori c:/xampp
2. Masuk ke dalam directori c:/xampp
3. Pilih file xampp-control.exe, kemudian jalankan dengan menekan <enter> atau klik dua kali, tampilan berikut akan ditampilkan :



(Sumber: Sidik 2012:73)

Gambar 2.4. Tampilan XAMPP Control Panel

Dari tampilan di atas, beri tanda cek pada baris *Svc Apache*, dialog konfirmasi unjuk menjalankan Apache menjadi *Service* ditampilkan seperti berikut :



(Sumber: Sidik 2012:73)

Gambar 2.5. Tampilan Dialog Konfirmasi Menjalankan Apache

4. Klik tombol **OK**.
5. Setelah itu klik tombol start pada baris yang sama, menjalankan Apache. Maka tampilan menjadi seperti berikut :



(Sumber: Sidik 2012:73)

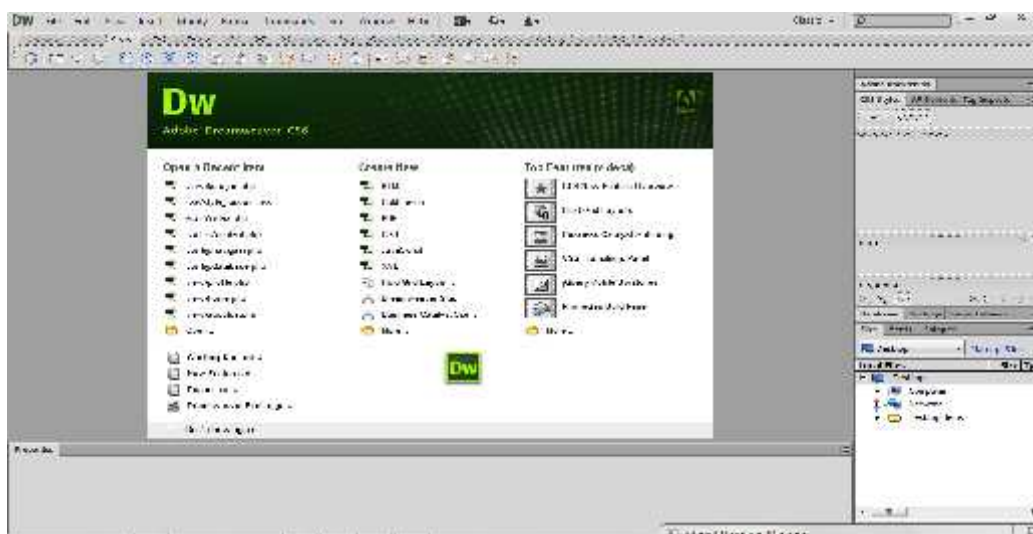
Gambar 2.6. Tampilan Svc Apache Running



2.4.6. Sekilas tentang Adobe Dreamweaver CS6

2.4.6.1. Pengertian Adobe Dreamweaver CS6

ELCOM (2013:1), “*dreamweaver cs6* adalah sebuah editor profesional yang menggunakan HTML untuk mendesain *web* secara *visual* dan mengelola situs atau halaman *web* dengan beberapa kemampuan. *Adobe Dreamweaver CS6* memiliki beberapa kemampuan. Versi ini bukan hanya *software* untuk desain *web* tetapi juga untuk menyunting kode serta pembuatan aplikasi *web* antara lain *JSP*, *PHP*, *ASP*, *XML*, dan *ColdFusion*.”



(Sumber : ELCOM, 2013:2)

Gambar 2.7. Tampilan Awal Adobe Dreamweaver CS6

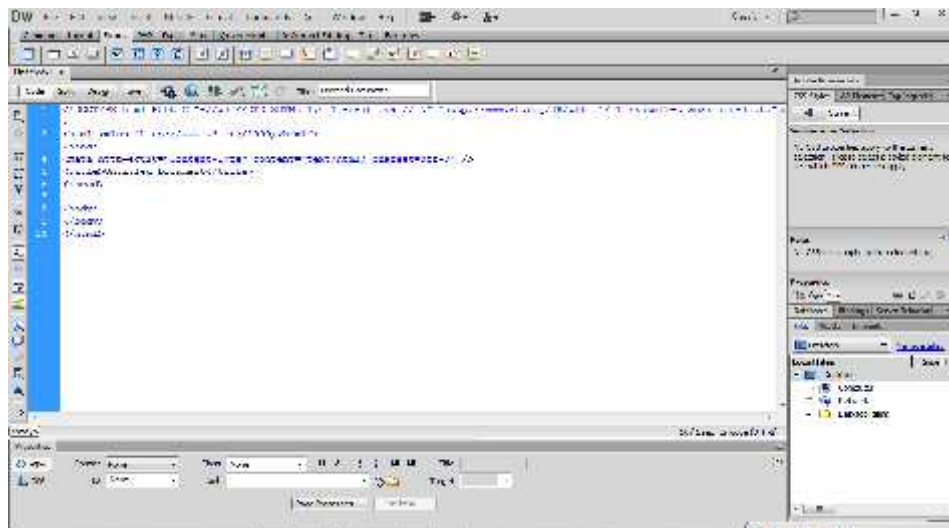
2.4.6.2. Ruang Kerja Adobe Dreamweaver CS6

ELCOM (2013:3) menjelaskan tentang ruang kerja *Dreamweaver CS6* adalah:

1. **Insert Bar**, berisi tombol-tombol untuk menyisipkan berbagai macam objek seperti *image*, *table*, dan *layer* ke dalam jendela dokumen.
2. **Document Toolbar**, berisi tombol-tombol dan menu pop-up yang menyediakan tampilan berbeda dari jendela dokumen.
3. **Coding Window**, berisi kode-kode *HTML* dan tempat untuk menuliskan kode-kode pemrograman, misalnya *PHP* atau *ASP*.
4. **Panel Group**, berisi kumpulan panel yang saling berkaitan satu sama lainnya yang dikelompokkan di bawah satu judul.
5. **Property Inspector**, digunakan untuk melihat dan mengubah berbagai properti objek atau teks.
6. **Jendela Dokumen**, digunakan untuk menampilkan dokumen saat dimana Anda sekarang bekerja.



7. **Ruler**, mempermudah ukuran dalam mendesain halaman *web*.
8. **Site Panel**, digunakan untuk mengatur *file-file* dan *folder-folder* yang membentuk situs *web* Anda.



(Sumber : ELCOM, 2013:2)

Gambar 2.8. Ruang Kerja Adobe Dreamweaver CS6

2.4.7. JavaScript

Sidik (2011:1), “JavaScript adalah salah satu pengetahuan yang penting dikuasai oleh pengembang program berbasis *web* karena dengan JavaScript ini dapat membuat dokumen HTML yang ditampilkan dalam *browser* menjadi lebih interaktif dengan memberikan beberapa fungsionalitas kedalam halaman *web* dengan menggunakan antarmuka (*interface*) *web*.

Winarno, dkk. (2014:129), “JavaScript adalah bahasa scripting client side yang sangat populer. Ini karena JavaScript bisa dipakai di HTML, web, untuk server, pc, laptop, tablet, ponsel dan lainnya.”

Winarno, dkk (2014:134) menjelaskan bahwa cara menggunakan JavaScript adalah dengan menggunakan tag `<script>` dan `</script>`. JavaScript biasanya dituliskan pada bagian `<body>` dalam HTML. Contohnya :



```
<!Doctype html>
<html>
<body>

<script>
document.write("<h1>Ini adalah Heading </h1>");
document.write("<p>Ini adalah pragraf.</p>");

</body>
</html>
```

(Sumber: Winarno, dkk., 2014:134)